IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
TYLER DIVISION

GEMALTO S.A., Plaintiff, v. HTC CORPORATION, HTC AMERICA, INC., EXEDEA, INC., SAMSUNG ELECTRONICS CO., LTD., SAMSUNG TELECOMMUNICATIONS AMERICA LLC, MOTOROLA MOBILITY, INC., and GOOGLE INC., Defendants.	Civil Action No. 6:10-CV-561-LED JURY TRIAL DEMANDED

DEFENDANTS' RESPONSIVE CLAIM CONSTRUCTION BRIEF

TABLE OF CONTENTS

Page

INTRODUCTION 1

BACKGROUND 1

I. THE SPECIFICATION DESCRIBES USING A HIGH LEVEL PROGRAMMING LANGUAGE WITH A MICROCONTROLLER. 1

II. THE ASSERTED CLAIMS 3

III. TO AVOID PRIOR ART, THE APPLICANTS REPEATEDLY DISTINGUISHED MICROPROCESSORS AND MICROCONTROLLERS. 4

IV. THE ACCUSED *MICROPROCESSOR* BASED PRODUCTS 5

ARGUMENT 5

HARDWARE TERMS 5

I. *MICROCONTROLLER* ('317 CLAIMS 58, 59, 62, 63, 65-69, 73, 75, 77-79, 81, 87-89, 91, 92; '485 CLAIMS 1, 2, 6-11, 14, 16, 18, 20-22, 25) 5

A. The Intrinsic Evidence Does Not Allow a Microprocessor to be a "Microcontroller." 6

B. The Parties' Extrinsic Evidence Supports Defendants' Construction. 8

C. Gemalto Fails To Demonstrate That A "Microcontroller" *Is* A "Microprocessor." 9

D. Gemalto's "Device That Includes" Language Should Be Rejected. 10

E. The Preamble Of Claim 7 Of The '485 Patent *Is* A Limitation. 11

II. *INTEGRATED CIRCUIT CARD* ('317 CLAIMS 1, 4-11, 13-15, 22, 24, 25, 30, 31, 55, 64, 84-86, 93, 94; '485 CLAIMS 38, 39, 40, 42, 43) 11

A. An Integrated Circuit Card Is A "Card," Not A "Base." 12

B. Defendants' "Not a Microprocessor System" Limitation Is Appropriate. 14

III. *PROGRAMMABLE DEVICE* ('727 CLAIMS 1-7, 10, 12, 14, 16-18) 14

A. The Intrinsic Evidence Mandates That a Programmable Device Is a Device Controlled by a Microcontroller. 15

B. Gemalto's Construction Has No Support. 16

C. Programmable Device *Is* a Limitation of Claim 3 of the '727 Patent. 16

IV. *RESOURCE CONSTRAINTS* ('317 CLAIMS 65, 78, 88, 91, 92; '485 CLAIMS 7, 21; '727 CLAIMS 3, 17)....16

A. "Resource Constraints" Is Indefinite. 16

B. Gemalto's Construction Does Not Resolve, But Rather Highlights, the Insoluble Ambiguity with "Resource Constraints."....17

V. *TERMINAL* ('317 CLAIMS 1, 10, 25, 30, 31, 40, 55, 59, 62, 64, 84-86, 93, 94; '485 CLAIMS 2, 38, 40, 42)....20

VI. *THE INTEGRATED CIRCUIT CARD* ('317 CLAIMS 58, 65, 78, 87, 88, 92)....22

SOFTWARE TERMS....22

I. *HIGH LEVEL [PROGRAMMING] LANGUAGE* ('317 CLAIMS 1, 5, 31, 35, 64, 65, 84-86, 93, 94; '485 CLAIMS 7, 38, 40, 42; '727 CLAIM 3)....22

II. *COMPILING / COMPILED FORM* ('317 CLAIMS 1, 31, 64-68, 73, 84-88, 91-94; '485 CLAIMS 1, 7-10, 14, 16, 21, 24, 38-43; '727 CLAIMS 1, 3-6, 10, 12, 17, 20) 23

III. *BYTE CODES / VIRTUAL MACHINES* ('317 CLAIMS 1, 31, 58, 64, 65, 78, 84-88, 91-94; '485 CLAIMS 14, 21, 24, 38-43; '727 CLAIMS 10, 17, 20) 25

IV. *CLASS FILE FORMAT* ('317 CLAIMS 58, 63, 67, 87; '485 CLAIMS 1, 6, 9; '727 CLAIMS 1, 2, 5)....26

V. *CONVERTING / CONVERTED FORM* ('317 CLAIMS 1, 31, 58, 64, 78, 84-88, 91- 94; '485 CLAIMS 1, 21, 38, 42; '727 CLAIMS 1, 10, 17, 20)....27

VI. *SPECIFIC BYTE CODES / GENERAL BYTE CODES* ('317 CLAIMS 1, 31, 58, 64, 65, 78, 84-87, 91, 94; '485 CLAIMS 14, 24, 39, 41, 43; '727 CLAIMS 10, 20)....29

MEANS PLUS FUNCTION CLAIMS 29

VII. *ATTRIBUTES* ('317 CLAIM 66; '485 CLAIM 8; '727 CLAIM 4) 30

CONCLUSION....30

TABLE OF AUTHORITIES

Page

Cases

Catalina Marketing Int'l, Inc. v. Coolsavings.com, Inc.,
289 F.3d 801 (Fed. Cir. 2002).....11

Globetrotter Software, Inc. v. Elan Computer Group, Inc..,
362 F.3d 1367 (Fed. Cir. 2004).....23

Hakim v. Cannon Avent Group, PLC et al.,
479 F.3d 1313 (Fed. Cir. 2007)..... 15-16

Halliburton Energy Servs., Inc. v. M-I LLC,
514 F.3d 1244 (Fed. Cir. 2008).....17

Iconfind Inc. v. Yahoo! Inc.,
2009 WL 8454648 (E.D.Cal. 2009).....10

iLOR, LLC v. Google, Inc.,
550 F.3d 1067 (Fed. Cir. 2008)..... 15-16

Interactive Gift Exp., Inc. v. Compuserve Inc.,
256 F.3d 1323 (Fed Cir. 2001).....12

NTP, Inc. v. Research In Motion, Ltd.,
418 F.3d 1282 (Fed. Cir. 2005).....11

PC Connector Solutions LLC v. SmartDisk Corp.,
406 F.3d 1359 (Fed. Cir. 2005).....19, 20

Phillips v. AWH Corp.,
415 F.3d 1303 (Fed. Cir. 2005).....7, 24

SciMed Life Sys., Inc. v. Adv. Cardio. Sys., Inc.,
242 F.3d 1337 (Fed. Cir. 2001).....8

Springs Window Fashions LP v. Novo Indus., L.P.,
323 F.3d 989 (Fed. Cir. 2003).....8, 15

Sta-Rite Industries, LLC v. ITT Corp.,
682 F. Supp. 2d 738 (E.D.Tex. 2010).....29

TorPham, Inc. v. Ranbaxy Pharmaceuticals,
336 F.3d 1322 (Fed. Cir. 2003).....8

Vitronics Corp. v. Conceptronic, Inc.,
90 F.3d 1576 (Fed. Cir. 1996).....23

Voice Tech. Group, Inc. v. VMC Sys., Inc.,
164 F.3d 605 (Fed. Cir. 1999).....9

WMS Gaming, Inc. v. International Game Technology,
184 F.3d 1339 (Fed. Cir. 1999).....30

White v. Dunbar,
119 U.S. 47 (1886).....8, 15

Statutes

35 U.S.C. § 112.....12

NOTE ON CITATIONS

- References to *Plaintiff's Opening Claim Construction Brief* (February 24, 2012) are indicated by the abbreviation "Br.," followed by the page number being cited. "Br. 5" therefore refers to page 5 of Plaintiff's opening brief.
- The patents-in-suit, U.S. Patent. Nos. 6,308,317, 7,117,485, and 7,818,727, are attached as Exhibits A, B, and C, respectively. References to the patents-in-suit are indicated by column and line number, or by claim number. Unless stated otherwise, all references are to the specification of the '317 patent. A reference to "3:15-17" therefore means column 3, line 15 through line 17 of Exhibit A, U.S. Patent No. 6,308,317.
- Portions of the prosecution histories of the '317, '485, and '727 patents are attached as Exhibits D, E, and F, respectively.
- Portions of the re-examination of the '317 patent are attached as Exhibit G.
- Depositions are referred to be the name of the deponent only. A reference to "Guthery, 4:12-25" therefore means pages 4, lines 12-15 of the Guthery deposition.
- Other exhibits are attached as Exhibits H through P.

INTRODUCTION

Defendants' proposed constructions are consistent with the ordinary and customary meaning of the terms at issue in the context of the intrinsic record. In contrast, in an attempt to preserve both its validity and infringement cases, Gemalto repeatedly contradicts its reasoning, offering different approaches to claim construction depending on its desired outcome.

For example, the intrinsic evidence and even Gemalto's own extrinsic evidence make clear that—as was commonly known in the art—the claimed "microcontrollers" are separate and distinct from "microprocessors." The applicants repeatedly explained the alleged invention by contrasting the claimed memory-constrained microcontrollers with microprocessors that are not so constrained. Gemalto's own extrinsic evidence makes the same distinction. Yet in a attempt to ensnare Defendants' accused microprocessor-based devices, Gemalto improperly asserts that the claims should somehow encompass microprocessors. At the same time, in an effort to preserve validity, Gemalto proposes that common terms like "converting" and "compiling" should be limited to only very narrow embodiments in the specification, contrary to their ordinary meanings. And it proposes a construction of "resource constraints" that evolves over time such that a device that at one point was not "resource constrained," could later be.

Defendants' constructions should be adopted and Gemalto's constructions rejected.

BACKGROUND

I. THE SPECIFICATION DESCRIBES USING A HIGH LEVEL PROGRAMMING LANGUAGE WITH A MICROCONTROLLER.

U.S. Patent Nos. 6,308,317 ("the '317 patent"), 7,117,485 ("the '485 patent"), and 7,818,727 ("the '727 patent") share the same specification, the same title—"Using a High Level Programming Language with a Microcontroller"—and claim priority to the same 1996 provisional application. (Br. 1.) The claimed invention describes a means for running programs

written in high-level computer programming languages, like Java, on the limited resources of a microcontroller or integrated circuit card. (1:16-19; 1:55-61; 2:46-28.)

As Gemalto concedes, at the time of the alleged invention high-level programming languages like Java already ran on microprocessor-based computers: Java "was designed to operate on general purpose computing platforms, *i.e.*, on 'microprocessor-based computers, with access to relatively large amounts of memory.'" (Br. 3 (*citing* 1:56-58); 2:5-7). A microprocessor can access large amounts of memory because it "is not constrained by what will fit on a single integrated circuit device," but rather can access memory on other "chips." (2:26-33.) Typically, a "microprocessor system [] has 1 to 128 megabytes of RAM." (2:26-30.)

The patents distinguish these prior art "microprocessor implementations" from implementations with microcontrollers, noting that "there are no conventional Java implementations on microcontrollers, as would typically be used in a smart card." (1:58-60.) Unlike microprocessors, microcontrollers typically contained applications written in "low-level" programming languages, which conserved memory at the expense of making programs more difficult to write. (2:45-47.) The limited memory on a microcontroller (below left) was a critical aspect of the alleged invention of the patents. As the patents observe, "[m]icrocontrollers differ from microprocessors in many ways. In contrast to the microprocessor, a microcontroller

includes a central processing unit, memory and other functional elements, all on a single semiconductor substrate, or integrated circuit (*e.g.*, a 'chip')" (2:2-5 (emphasis added); Br. 13.) Thus, "[i]n a microcontroller, the amount of each kind of memory available is constrained by the amount of space on the integrated circuit used for each kind of

memory." (2:14-16.) "A typical microcontroller can access one to sixty-four kilobytes of built-in memory, with sixteen kilobytes being very common." (2:8-10; 2:26-34: "a microcontroller typically has a small RAM of 0.1 to 2.0 K, 2K to 8K of EEPROM, and 8K-56K of ROM.") Due to their small size, microcontrollers are and were often used on integrated circuit cards, also known as "ICCs" or "smart cards." The bank card (above) is an example of an ICC, with the microcontroller above the first "3333" digits of the card. Microcontrollers may also control devices like automobile engine parts (but not an entire car). (18:31-34.)

In order to deal with the problem the patents seek to solve—the microcontroller's vastly limited memory resources—the patents disclose a "virtual machine" which was small enough to operate on microcontrollers. (8:20-22.) The specification also recites a "converter" that converts applications developed for a regular Java virtual machine into a form that could run on the size-reduced virtual machine recited in the specification. (8:35-50; Fig. 2.)

II. THE ASSERTED CLAIMS

Each of the 100 asserted claims require applications that are "compiled" then "converted" to run on the limited resources of a microcontroller, integrated circuit card, or programmable device. For example, claim 1 of the '485 patent recites a "microcontroller":

1. A microcontroller comprising:

a memory storing

> a derivative application derived from an application having a class file format wherein the application is derived from an application having a class file format by first compiling the application having a class file format into a compiled form and then converting the compiled form into a converted form, and
>
> an interpreter configured to interpret derivative applications in the converted form and derived from applications having a class file format; and

a processor coupled to the memory, the processor configured to use the interpreter to interpret the derivative application for execution.

Claim 1 of the '317 patent requires an "integrated circuit card" and includes similar compiling/converting limitations, while claim 1 of the '727 patent requires a "programmable device," again with similar compiling/converting elements. The 76 asserted dependent claims add various known features and functions of microcontrollers, ICCs, and the Java programming language, or require that the "high level programming language" required by some of the claims be the Java language itself (*e.g.*, claim 5 of the '317 patent, claim 2 of the '727 patent).

III. TO AVOID PRIOR ART, THE APPLICANTS REPEATEDLY DISTINGUISHED MICROPROCESSORS AND MICROCONTROLLERS.

During the prosecution of the asserted patents, the applicants traversed prior art by explicitly relying on the same distinctions between microprocessors and microcontrollers, and the memory available to each. Again and again, the applicants asserted that the point of novelty of the alleged invention was putting a memory-intensive high level language designed for use with microprocessor systems, on a memory-constrained microcontroller or smart card.

For example, while the specification notes a microcontroller would typically have 0.1 to 2.0 kilobytes of RAM, the applicants stated to the PTO that, "[a]t the time of the invention, the typical Java Virtual Machine required over 1 MB of memory. Any person of ordinary skill would realize that to squeeze such an interpreter into an integrated circuit card (such as a smart card) is anything but an obvious task." (Ex. D at GEM2983.) When the applicants sought to distinguish the Peyret reference after rejections of all the '317 and '485 claims, the applicants argued that "Peyret does not deal with a solution of how to squeeze Java or another high level language onto a smart card" and thus was not an enabling reference. (Ex. D at GEM3029 (emphasis added); *see also* Ex. E at GEM3766 ("Peyret is completely silent on how to enable a program written in a high level language to operate on an integrated circuit card.").) The

challenge presented by these dramatically reduced resources made "fitting Java technology inside smartcards [] like playing golf in a phone booth." (Br. 3; *see also* Ex. D at GEM3030-32.)

During the re-examination of the '317 patent, Gemalto again pointed out the importance of the memory constraints of microcontrollers in contrast to microprocessors. Gemalto noted that "the Java runtime environment has a minimum of 32 MB bytes [sic] of memory, 125 MB of free disk space," whereas "in 1996 smart cards only had 512 bytes [] of RAM…. The ratio between 32 MB and 512 bytes is 65536." The alleged 'solution' of the patent was to "deal with the resource constraints of smart cards." (Ex. G at GEM4517; *see generally id.* at GEM4515-17.)

IV. THE ACCUSED *MICROPROCESSOR* BASED PRODUCTS

The accused products comprise various smartphones and tablets using the Android platform, such as the Nexus One phone shown on the left. These products are general-purpose computers that employ microprocessors. Unlike "a small RAM of 0.1 to 2.0" that the patents attribute to microcontrollers (2:39-41), the accused devices have between 117MB and 1GB of RAM: at least 50,000 times more memory. This memory is at least as much, and usually far exceeds, the 1 to 128 megabytes of RAM the patents attribute to microprocessors. (2:33-39.) It also far exceeds the 32 MB needed for the Java Run Time environment, whose size the applicants deemed prohibitive to running Java on a microcontroller-based system. (Ex. G at GEM4515-17.)

ARGUMENT

HARDWARE TERMS

I. *MICROCONTROLLER* ('317 CLAIMS 58, 59, 62, 63, 65-69, 73, 75, 77-79, 81, 87-89, 91, 92; '485 CLAIMS 1, 2, 6-11, 14, 16, 18, 20-22, 25)

Defendants	*Gemalto*
a single semiconductor substrate or integrated circuit designed specifically for embedded applications that	a device designed specifically for embedded applications that includes a

includes a central processing unit, memory and other functional elements, and that does not require external memory to function properly; not a microprocessor	central processing unit, memory and other functional elements on a single semiconductor substrate, or integrated circuit

The parties agree that microcontrollers have a "single semiconductor substrate or integrated circuit" that contains "a central processing unit, memory and other functional elements." They also agree that microcontrollers are "designed specifically for embedded applications." (Br. 13.) The parties' dispute is whether it is appropriate to allow a construction for "microcontrollers" that could include microprocessors, which would be inconsistent with ordinary meaning of the terms, the intrinsic evidence that consistently and explicitly contrasts the two in describing the alleged invention, and the relevant extrinsic evidence cited by both parties.

A. The Intrinsic Evidence Does Not Allow a Microprocessor to be a "Microcontroller."

As the parties agree, a microcontroller already has "a central processing unit, memory and other functional elements," all on a single semiconductor substrate or integrated circuit. In other words, the memory used by a microcontroller is on the microcontroller itself. Thus, as Defendants' construction appropriately provides and for which there really should be no dispute, a microcontroller does not require external memory to function properly. Indeed, the patents specifically contrast a microprocessor, whose central processing unit "requires certain external components (*e.g.*, memory, input controls and output controls) to function properly," from a microcontroller on this basis. (1:63-66) (emphasis added).

Nor should there be any dispute that Defendants' "not a microprocessor" construction is appropriate. The whole point of the alleged invention centers on the fact that, as the specification states, "[m]icrocontrollers differ from microprocessors in many ways." (1:62-63) Specifically, "[a]s compared to the relatively large external memory accessed by the microprocessor," on which high level languages were already indisputably used, "the typical

microcontroller accesses a much smaller memory" because it has its "functional components like memory on a single semiconductor substrate."[1] (*See generally* 1:55-2:34.) Indeed, if a "microcontroller" were a type of "microprocessor" as Gemalto asserts, the problem that the alleged invention addressed—"using a high level programming language with a microcontroller" (the title of each asserted patent)— would have already been solved: high level languages already worked on microprocessors. As a construction that would allow a microprocessor to meet the "microcontroller" limitations would nullify the alleged invention, it should be rejected. *Phillips v. AWH Corp*., 415 F. 3d 1303, 1315 (Fed. Cir. 2005) ("the specification is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term.") (emphasis added).

To traverse the prior art during prosecution, the applicants similarly and repeatedly described the claimed invention as allowing high-level programming languages developed for microprocessors to function on microcontrollers as well. (Ex. E at GEM3725 ("Applicants recognized the difficulty in operating Java (or other high level language) programs within the limited resources of an integrated circuit card or other microcontroller."); *see also* Ex. D at GEM2983-84, GEM3028-32 ("Peyret does not teach or suggest Applicants' novel and non-obvious invention of how programs written in JAVA or other high level programming languages may be executed on a smart card."); Ex. E at GEM3763-66 ("Appellants concede these steps [to compile and run Java] to be in the prior art. . . . At the time of the invention, the typical Java Virtual Machine required over 1 MB of memory. Any person of ordinary skill would realize that to squeeze such an interpreter into an integrated circuit card (such as a smart card) is anything

[1] While Plaintiff argues other statements relating to the microprocessor use qualifying language such as "typical," (Br. 15), this language refers to "typical" memory ranges of the art the time, and in no way qualifies the critical differences between the microprocessor and microcontroller detailed in the provisional and the remainder of the intrinsic evidence.

but an obvious task."); Ex. G at GEM4515-19 ("The Java run time environment, the JRE, requires a system that has a minimum of 32MB of memory, 125 MB of free disk space. However, in 1996 smart cards only had 512 bytes, not kilobytes or megabytes, just 512 bytes of RAM and 4K bytes of EEPROM.").) Defendants' construction holds the applicants to their statements that the alleged novelty of the claimed invention is placing a high-level programming language on a microcontroller, as opposed to microprocessor systems which the applicants conceded could already run high level languages.[2] *White v. Dunbar*, 119 U.S. 47, 51-52 (1886) (patent claim is not a "nose of wax" to be twisted one way to preserve a patent's validity and another way to catch an alleged infringer); *Springs Window Fashions LP v. Novo Indus., L.P.*, 323 F.3d 989, 995 (Fed. Cir. 2003) ("A patentee may not state during prosecution that the claims do not cover a particular device and then change position and later sue a party who makes that same device for infringement."); *SciMed Life Sys., Inc. v. Adv. Cardio. Sys., Inc.*, 242 F.3d 1337, 1343 (Fed. Cir. 2001) (holding that since the patents distinguished the prior art dual lumen devices, the claims should not be read so broadly as to encompass dual lumens); *TorPham, Inc. v. Ranbaxy Pharmaceuticals*, 336 F.3d 1322, 1329 (Fed. Cir. 2003 (a patentee may not "adopt a position contradictory to that adopted before the PTO and expect to be believed.").

B. The Parties' Extrinsic Evidence Supports Defendants' Construction.

Gemalto's cited extrinsic evidence further supports Defendants' construction that a microcontroller cannot be a microprocessor. For example, The Microcontroller Idea Book notes that "both microprocessors and microcontrollers contain a central processing unit." (Dkt. 163, Ex. M at 2.) But like the patents, the book explains that a microprocessor requires and can access external memory, while a microcontroller is limited to the memory on a single chip:

[2] As with most terms, Gemalto ignores Defendants' voluminous disclosed evidence as to "microcontroller." Any attempt by Gemalto to address this evidence for the first time on reply, when Defendants have no opportunity to respond in writing, should be ignored.

> To make a complete computer, a microprocessor requires memory for storing data and programs, and input/output (I/O) interfaces for connecting external devices like keyboards and displays.
>
> In contrast, a microcontroller is a single-chip computer because it contains memory and I/O interfaces in addition to the CPU. Because the amount of memory and interfaces that can fit on a single chip is limited, microcontrollers tend to be used in smaller systems that require little more than the microcontroller and a few support components.

Id. at 2 (emphasis added); *id.* at 1 ("A microcontroller is a computer-on-a-chip, or, if you prefer, a single-chip computer.")). Thus while a microprocessor cannot function without external memory, a microcontroller "is a single chip computer" that, as Defendants' construction provides, can function properly without external memory, and is not a microprocessor. Inventor Wilkinson agreed that microprocessors and microcontrollers differed. (Wilkinson, 25:6-8 ("Q: You would agree that a microprocessor and a microcontroller aren't the same thing? A: I don't think they are the same thing."); *see also id*. at 125:6-16.) An inventor is "a competent witness to explain the invention and what was intended to be conveyed by the specification and covered by the claims." *Voice Tech. Group, Inc. v. VMC Sys., Inc.*, 164 F.3d 605, 615 (Fed. Cir. 1999).

By clearly stating what a microcontroller is and what it is not, Defendants' construction cleanly conveys the understanding of one of skill in the art to the jury and should prevail.[3]

C. Gemalto Fails To Demonstrate That A "Microcontroller" *Is* A "Microprocessor."

Despite the repeated and consistent distinction between microcontrollers and microprocessors in the intrinsic evidence, Gemalto asserts that "a microcontroller is a special type of microprocessor." (Br. 15.) Gemalto points to nothing in the claims, specifications or prosecution histories of the asserted patents to support this assertion.

[3] Notably, in discussing its construction for the "resource constraints" limitation, even Gemalto distinguishes microprocessors and microcontrollers. (Br. 17-18.)

Instead, Gemalto only cites to the provisional application, which refers to a "microprocessor card," a microprocessor "component" or "hardware" of an ICC, and a figure that refers to a microprocessor. (Br. 15). These references are plainly meant to refer to the CPU, *i.e.*, the processor, on the ICC.[4] Indeed, like the remainder of the intrinsic evidence, the provisional application makes clear that microprocessors are not microcontrollers:

> Microcontrollers differ from microprocessors in many ways. A microprocessor is a central processing unit without any memory. The microprocessor couples to a printed circuit board which includes memory, input and output controls and all the other necessary and desired functional elements.... A microcontroller comprises a central processing unit, memory and other functional elements on a single chip.

(Dkt. 163, Ex. Q at 1:14-24) (emphasis added).) For the same reasons discussed above in connection with the specification and prosecution history, this language in the provisional would make no sense if a microcontroller were the same as a microprocessor. Notably, the quote above from the provisional that "microprocessors differ from microcontrollers in many ways" language was kept in the specification. The language Gemalto attempts (unsuccessfully) to rely upon was not. *Iconfind Inc. v. Yahoo! Inc*., 2009 WL 8454648, 11 (E.D.Cal. 2009) (holding that while provisional patent applications may have used a term to express multiple meanings, the patent claims and specification used a consistent meaning and that meaning controlled).

D. Gemalto's "Device That Includes" Language Should Be Rejected.

Gemalto's construction improperly defines a microcontroller as "a device... that includes a central processing unit, memory and other functional elements on a single semiconductor substrate, or integrated circuit." This improperly blurs the distinction between a microcontroller and a device that contains a microcontroller. The phone, key ring, ring, and car depicted in Figs.

[4] Gemalto apparently seeks to obscure the distinction between a *processor* and a *microprocessor*, treating the two as interchangeable. (Br. 1.) They are not. As the parties agree, microcontrollers contain a central processing unit or "CPU" (Br. 13), *i.e.* a processor. It would make no sense for a microcontroller to both contain a "microprocessor" along with other components while simultaneously being a type of microprocessor, as Gemalto asserts.

22-25 all include a "a central processing unit, memory and other functional elements on a single semiconductor substrate." (18:31-42.) But these items are not themselves microcontrollers. Gemalto's ambiguous language appears aimed to achieve wiggle room to improperly argue that a smartphone can somehow be a microcontroller merely because it may "include" a CPU, memory and other components. Gemalto's litigation-inspired construction is contrary to the ordinary meaning of microcontroller and the intrinsic evidence, and should be rejected.

E. The Preamble Of Claim 7 Of The '485 Patent *Is* A Limitation.

In a footnote, Gemalto concludes without argument that "microcontroller" is not a limitation of claim 7 of the '485 Patent. (Br. 13, FN 6.) "Whether to treat a preamble as a limitation is a determination resolved only on review of the entire patent to gain an understanding of what the inventors actually invented and intended to encompass by the claim." *Catalina Marketing Int'l, Inc. v. Coolsavings.com, Inc.*, 289 F.3d 801, 808 (Fed. Cir. 2002); *NTP, Inc. v. Research In Motion, Ltd.*, 418 F.3d 1282, 1305 -1306 (Fed. Cir. 2005).) Further, "clear reliance on the preamble during prosecution to distinguish the claimed invention from the prior art transforms the preamble into a claim limitation because such reliance indicates use of the preamble to define, in part, the claimed invention." *Catalina Marketing*, 289 F.3d at 808. As detailed above, the applicants repeatedly defined the claimed invention as operating on microcontrollers and distinguished the claimed microcontrollers from microprocessors in traversing prior art. Thus, the "microcontroller" element of the preamble of claim 7 of the '485 patent is a limitation, and Gemalto provides no argument to the contrary.

II. *INTEGRATED CIRCUIT CARD* ('317 CLAIMS 1, 4-11, 13-15, 22, 24, 25, 30, 31, 55, 64, 84-86, 93, 94; '485 CLAIMS 38, 39, 40, 42, 43)

Defendants	*Gemalto*
a card containing a central processing unit, memory and other functional elements, all on a single semiconductor substrate or integrated circuit, that does not require external memory to	an integrated circuit containing a central processing unit, memory and

function properly; not a microprocessor system; OR a card containing a microcontroller	other functional elements on a base

The parties dispute whether an "integrated circuit card" is in fact a "card," as the claims themselves recite, or whether it is a "base" as Gemalto asserts to seek wiggle room to improperly argue that motherboards (a frame with connections between slots for placing various components) in Defendants' devices are somehow ICCs. The parties also again dispute whether ICCs can encompass microprocessor systems.

A. An Integrated Circuit Card Is A "Card," Not A "Base."

As the plain language in the claims provides, an "integrated circuit card" is a "card." *Interactive Gift Exp., Inc. v. Compuserve Inc.*, 256 F.3d 1323, 1331 (Fed Cir. 2001) ("In construing claims, the analytical focus must begin and remain centered on the language of the claims themselves, for it is that language that the patentee chose to use to 'particularly point[ ] out and distinctly claim[ ] the subject matter which the patentee regards as his invention.'" (quoting from 35 U.S.C. § 112, ¶ 2)). The specifications further explain that an "integrated circuit card" is "a plastic card that has substantially the same dimensions as a typical credit card" with an embedded integrated circuit. (7:57-59; 2:35-3:46 (contextualizing ICCs with "card" terms, *i.e.*, "smart cards," "contact- based" and "contactless cards," "debit cards," ATM access cards, "card users" and "security of the card").) Microcontrollers are often the "integrated circuit"—with "the CPU, memory, and other functional elements" required by both parties' constructions—on an integrated circuit card as the specification recites. (2:35-37.) Named inventor Guthrey further confirmed that an integrated circuit card is "a card complying with the ISO 7816 series of standards." (Guthrey at 7:17-19; *see generally id.* at 7:17 – 8:21.)

Nevertheless, Plaintiff seeks to rewrite the claim language by replacing the word "card" with the word "base." There is no reason to do so. "Base" implies a support structure that is not a "card," *e.g.*, the base of a trophy. (*See, e.g.*, Ex. H: "base: the bottom of something considered

as its support: that on which something rests or stands: foundation"). This bears no relation to the integrated circuit cards consistently and solely disclosed in the intrinsic evidence.

Gemalto's only purported support that an "integrated circuit card" is somehow not a "card" is the following incomplete quote from the specification**:** "[i]n other embodiments, ... *mounted within bases* other than a plastic card." (Br. 16 (citing 7:59-62) (emphasis added by Gemalto).) But the full quote reveals that the specification distinguishes ICC embodiments from "other embodiments" where the microcontroller is mounted within a base instead of a card:

> In some embodiments, the microcontroller, memory and communicator are embedded in a plastic card that has substantially the same dimensions as a typical credit card. *In other embodiments*, the microcontroller, memory and communicator are *mounted within bases other than a plastic card*, such as jewelry (*e.g.*, watches, rings or bracelets), automotive equipment, telecommunication equipment (*e.g.*, subscriber identity module (SIM) cards), security devices (*e.g.*, cryptographic modules) and appliances.

(7:57-65; *c.f.* Br. 16 (Gemalto's quoted language in italics).) This in no way suggests that an integrated circuit card can be a "base," but rather shows just the opposite.

Similar to its improper use of "device" in construing microcontroller, Gemalto also seeks to blur the distinction between an integrated circuit and integrated circuit card, in an attempt to sweep motherboards into the scope of the claims. Gemalto defines the integrated circuit card as an "integrated circuit" with certain features. But the integrated circuit card is a "card" with an integrated circuit. In other words, "integrated circuit" qualifies "card," not the reverse.

Moreover, despite its agreement that a microcontroller "includes a central processing unit, memory and other functional elements on a single semiconductor substrate, or integrated circuit" (Br. 13), and despite its concession that the integrated circuit card claims "narrowly refer to a specific type of device that includes a microcontroller" (Br. 28), Gemalto's construction of "integrated circuit card" omits any reference to "a single semiconductor substrate." Gemalto's omission is likely yet another attempt to sweep motherboards into the scope of "integrated circuit

card," even though unlike "a silicon substrate for chip" that has its components built in when manufactured, "a motherboard is something you could attach things to after the fact." (Wilkinson, 149:25 – 150:12.) For this reason as well, Gemalto's construction must be rejected.

B. Defendants' "Not a Microprocessor System" Limitation Is Appropriate.

As Gemalto concedes, "[t]he Integrated Circuit Card Clams [] narrowly refer to a specific type of device that includes a microcontroller and can use high level programming language – *e.g.*, an integrated circuit card." (Br. 27-28 (emphasis added).) Thus, the limitations concerning microcontrollers in Defendants' construction are appropriately included in the construction of ICC, including that ICCs do not require external memory to function properly and that an ICC cannot be a microprocessor system.[5] And as shown above, the claims make no sense if applied to devices utilizing microprocessors, as the alleged novelty of the patents was implementing high level languages—which already functioned on microprocessors—on resource-constrained microcontrollers. Assuming the Defendants' construction of "microcontrollers" is adopted, however, "a card containing a microcontroller" would be an acceptable alternative construction to Defendants' more lengthy construction that includes its definition of microcontroller.

III. *PROGRAMMABLE DEVICE* ('727 CLAIMS 1-7, 10, 12, 14, 16-18)

Defendants	*Gemalto*
integrated circuit card or other device controlled by a microcontroller; not a microprocessor-based system	No construction necessary. Alternate: a device that can execute a computer program

The parties dispute whether "programmable device" can include literally any computer or mechanical device that can execute a program (Plaintiff), or whether the applicants' made-up term should be construed in accordance with the intrinsic evidence (Defendants).

[5] Gemalto argues the language in the specification that "[t]he processor [on an ICC] may be a microcontroller" (Br. 16; 4:12-13) means the processor need not be a microcontroller. But this does not mean that the processor on an ICC could somehow be the same microprocessor applicants consistently distinguished in explaining the alleged invention.

A. The Intrinsic Evidence Mandates That a Programmable Device Is a Device Controlled by a Microcontroller.

"Programmable device" is not a term of art. Gemalto does not assert otherwise. Indeed, the named inventors confirmed the term had no meaning to them. (*See* Guthery, 155:2-7 ("Q. Does [programmable device] have any meaning to you? A. It has no meaning."); Wilkinson, 120:6-10 ("Q. At the time that you were working on the Java Card project, did the term "programmable device" have any particular meaning to you? A. Programmable device. I don't remember anybody ever talking about a programmable device particularly."))

Rather, "programmable device" refers to devices that the specification describes as containing embedded microcontrollers, such as telephones, key rings, jewelry, and the electrical subsystems of automobiles. (2:42-45; 7:57-65; 19:5-30; Figs. 22-25.):

> In other embodiments, the above-described techniques are used with a *microcontroller* (such as the processor 12) *may control devices* (*e.g.*, part of an automobile engine) other than an integrated circuit card. In these applications, the microcontroller provides a small platform (*i.e.*, a central processing unit, and a memory, both of which are located on a semiconductor substrate) for storing and executing high level programming languages. Most existing *devices* and new designs that utilize a *microcontroller* could use this invention to provide the ability to *program the microcontroller* using a high level language, and application of this invention to such devices is specifically included.

(18:34-42 (emphasis added)). As with "microcontroller" and "ICC," construing "programmable device" as "not a microprocessor-based system" is necessary to hold the applicants to the above-described representations to the Patent Office which repeatedly emphasized that the point of novelty of the alleged invention was running high-level language applications on an ICC or other microcontroller-based device, as opposed to the microprocessor systems that already ran such languages.[6] *Springs Window Fashions LP*, 323 F.3d at 995; *White v. Dunbar*, 119 U.S. at 51-52.

[6] The '727 claims as originally drafted required a "smart card," *i.e.*, an integrated circuit card. The Examiner examined and rejected the claims under non-statutory double patenting. Then the Applicants amended, without comment, all claims of the '727 patent to require a

B. Gemalto's Construction Has No Support.

Gemalto's construction of "device that can execute a computer program" seeks to expand the scope of the claims to include literally any device that can execute computer programs, including microprocessor or based systems, supercomputers, cars, televisions, MP3 players, airplanes, punch card machines, etc. Gemalto provides no support in the intrinsic record, but instead again ignores the applicants' repeated distinctions drawn to microprocessor-based systems. Its overly broad construction should be rejected.

C. Programmable Device *Is* a Limitation of Claim 3 of the '727 Patent.

Plaintiff asserts, again by footnote without argument, that "programmable device" is not a limitation of claim 3 of the '727 patent. As detailed above, the Applicants repeatedly relied on the microcontroller aspects of the programmable device in the specification and in prosecuting the claims, and should not be permitted to retreat from those representations.

IV. *RESOURCE CONSTRAINTS* ('317 CLAIMS 65, 78, 88, 91, 92; '485 CLAIMS 7, 21; '727 CLAIMS 3, 17)

Defendants	*Gemalto*
indefinite	computing resources that are limited when compared to conventional computing platforms, such as microprocessor-based desktop and personal computers

A. "Resource Constraints" Is Indefinite.

As detailed in Defendants' letter brief (Dkt., 158), the Asserted Patents do not define the term "resource constraints" or provide a way in which to determine the metes and bounds of the

"programmable device" rather than a "smart card." (Ex. F.) After the Applicants successfully filed a terminal disclaimer, the Examiner granted all claims in the '727 patent. When the examiner "allow[s] the continuation claims without further prosecution," it is presumed that allowance "was based on the prosecution argument in the parent." *Hakim v. Cannon Avent Group, PLC et al.*, 479 F.3d 1313, 1317 (Fed. Cir. 2007); *see also iLOR, LLC v. Google, Inc.*, 550 F.3d 1067, 1074-75 (Fed. Cir. 2008). If the applicants intended the "programmable device" claims of the '727 patent to include the microprocessors distinguished by the "microcontroller" and "integrated circuit card" claims in the '317 and '485 patents, then they failed to inform the Examiner as required. Applicants should not be permitted to sneak an expanded claim scope by the Examiner, and thus remain bound to their disclaimer of microprocessors.

term. Because the boundaries of "resource constraints" are not discernable they are indefinite: "a skilled artisan could not discern the boundaries of the claim based on the claim language, the specification, and the prosecution history, as well as her knowledge of the relevant art area." *Halliburton Energy Servs., Inc. v. M-I LLC*, 514 F.3d 1244, 1249-50 (Fed. Cir. 2008).

The specification of the Asserted Patents only states that "[a] microprocessor system typically has relatively little ROM and EEPROM, and has 1 to 128 megabytes of RAM." (2:25-34.) The specification then notes, "[a]s compared to the relatively large external memory accessed by the microprocessor, the typical microcontroller accesses a much smaller memory. A typical microcontroller can access one to sixty-four kilobytes of built-in memory, with sixteen kilobytes being very common." (2:5-10.) But this provides no guidance as to the metes and bounds of what memory, whether in terms of ROM, EEPROM or RAM, a device must have to qualify as "resource constrained." Indeed, named inventor Guthery indicated he would have no way of knowing whether any computer, including his own smart phone, was "resourced constrained" or not. (Guthery, 29:5-8; *see also id.* at 117:9 – 118:8 (claiming no knowledge of whether 100 MB of RAM is "resource constrained").)

B. Gemalto's Construction Does Not Resolve, But Rather Highlights, the Insoluble Ambiguity with "Resource Constraints."

1. Gemalto's construction is itself indefinite

Gemalto proposes that resource constraints be defined as "computing resources that are limited when compared to conventional computing platforms, such as microprocessor-based desktop and personal computers." This construction only highlights the insoluble ambiguity of "resource constraints." One of ordinary skill would not know what "limited computing resources" means, which resources should be compared with those of "conventional" computing platforms, or how many resources would avoid the "resource constraints" limitation.

Further, what is "conventional" to one person may not be "conventional" to another. Rather, the amount of memory available on a desktop or personal computer varies depending on how much money a person is willing to spend, or what she wants to use it for. This prevents saying any particular amount of memory is "conventional." (Guthery, 27:4-6 ("Q.Do you know how much RAM a conventional desktop computer has today? A. No, I don't); *id.* at 25:21 – 29:8; Wilkinson, 76:5-8 ("Q. So do you know what an appropriate range for what a conventional amount of RAM would be on a personal desktop computer today? A. I don't. I don't. I just know what I have.") (after noting that computer memory varies according a user's needs).)

Even if the amount of memory in a "conventional" desktop were firmly defined—say 2 gigabytes of RAM—Gemalto's construction would still be insolubly ambiguous. One of skill in the art would still not know how many less GB of RAM would be considered "limited" in comparison. 1.9 GB? 1.5 GB? 1.0 GB? 0.1 GB? 20KB?

Gemalto's construction also raises a false distinction based on microprocessors in "desktop and personal computers." As discussed above, in the specification and during prosecution, Applicants repeatedly and consistently distinguished microcontrollers from microprocessors as a whole. Applicants did not distinguish between microprocessors on "desktops and personal computers" and microprocessors on other devices, such as laptops or smartphones, which may themselves be considered "personal computers."

Indeed, the use of "personal computers" only presents further insolvable ambiguity. Inventor Wilkinson thought a "personal computer" was defined by whether it can be programmed. (Wilkinson, 150:13-23.) Inventor Guthery had no idea what the term meant at all. (Guthery, 148:14-21.). Further, Gemalto claims that an ICC, that it agrees contains a microcontroller, is "the smallest and most personal of computers." (Br. 2.) But Gemalto also

argues "a microcontroller is a type of microprocessor." (Br. 13-15). Following these positions to their logical conclusion, an ICC could itself be a non-resource constrained device because, according to Gemalto, it is a "personal computer" and it contains a microcontroller, which Gemalto contends is a type of microprocessor. In other words, under Gemalto's other arguments, an ICC would be a microprocessor-based personal computer. Gemalto cannot have it both ways, distinguishing microprocessors from microcontrollers for purposes of defining "resource constraints," while also arguing that microcontrollers are a type of microprocessor.

2. The scope of Gemalto's construction changes over time.

Gemalto's proposed construction purports to define "resource constraints" based on how one of ordinary skill in the art would understand memory constraints by a comparison to "conventional computing platforms" *today*.[7] Under Gemalto's construction, a top-of-the-line desktop computer, purchased in 1997 that met all the other elements of the claims, would not infringe in 1997 because its "resources" would not be "limited when compared to conventional computing platforms"—*i.e.*, other desktop computers in 1997. Presumably, however, this same device would infringe at some later point as "conventional computing platforms" became more powerful, thus making its memory "limited" compared to desktop computers sold in, say 2011.

The Federal Circuit considered and rejected such efforts at an evolving construction in *PC Connector Solutions LLC v. SmartDisk Corp.*, 406 F.3d 1359 (Fed. Cir. 2005). In that case, the court found terms "normally connectible," "conventional," "traditionally connectable," and

[7] Gemalto's construction is further infirm insofar as it is not even related to the relevant resource issues raised in the patent. As Gemalto admits, "the patentees contrasted those platforms that had sufficient resources to run Java-based programs and those that do not." (Br. 17.) Thus, to the extent there were any way to put a meaning to "resource constraints" at all, they would need to be tied to the "insufficient resources to run Java-based programs." But Gemalto seeks to compare devices to "conventional computing platforms" in the absence of any problem at all. (*C.f.* Guthery, 117:9 – 118:8, describing "resource constrained" as a technical parameter that depends on the problem being solved.)

"standard" as used in the claims were implicitly time-dependent. As a result, the scope of the claim limitations qualified by those terms was properly limited to technologies existing at the time of the invention. *Id.* at 1361-64. In reaching this decision, the Federal Circuit explained that "[a] claim cannot have different meanings at different times; its meaning must be interpreted as of its effective filing date." *Id.* at 1363 (citing *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 986 (Fed. Cir. 1995)). In other words, whether a device would be considered "resource constrained" as compared to today's "conventional computing platforms" as Gemalto's construction provides is not a proper manner to construe the claims as a matter of law.

Gemalto's attempt to distinguish *PC Connector* is unpersuasive. As Gemalto concedes, the Federal Circuit held in *PC Connector* that a term requiring a "conventionally connectable" I/O port is limited to what one of skill would consider "conventionally connectable" at the time the patent was filed in 1988. (Br. 20.) Here, Gemalto seeks to ensnare what would be "conventional" (whatever that means) today. The remaining authorities Gemalto relies upon are irrelevant because none stand for the proposition that the Federal Circuit rejected in *PC Connector, i.e.*, the improper assertion that a claim can have different scopes at different times.[8]

V. *TERMINAL* ('317 CLAIMS 1, 10, 25, 30, 31, 40, 55, 59, 62, 64, 84-86, 93, 94; '485 CLAIMS 2, 38, 40, 42)

Defendants	*Gemalto*
system, external to the microcontroller or ICC system, that communicates with the microcontroller or ICC system	a device that communicates with the integrated circuit card or microcontroller

[8] Gemalto's extrinsic evidence of unrelated patents that contain the terms "resource constraints" (Br. 19-20), is irrelevant to whether those terms are definite in this case. That these terms exist in other unrelated patents—with entirely different inventions, claims, specifications, and file histories—does not make the term in the patents-in-suit any more definite.

The parties agree that a terminal communicates with the ICC or microcontroller. They dispute, however, whether a "terminal" must be a separate system from the microcontroller or ICC (Defendants), or whether it can be any device that so communicates (Plaintiff).[9]

The intrinsic record supports Defendants' construction. In particular, the patents recite that a terminal communicates with the ICC, requiring that the terminal be separable from the ICC. (4:59-64) As the patents recite, "[t]erminals can be automated teller machines (ATMs), point-of-sale terminals, door security systems, toll payment systems, access control systems, or any other system that communicates with an integrated circuit card or microcontroller." (8:15-19.) Indeed, every described system in that list is external to the system that contains the ICC or microcontroller. The terminal is also consistently depicted as external to the ICC and its corresponding microcontroller in the figures. (Fig. 1, Fig. 13.) And in the provisional application, the terminal is defined as a "dedicated processing unit incorporating a card reader." (Ex. I at GEM4393.) Named inventor Guthrey also agreed that, in the context of an ICC, a terminal "would be something separate from the integrated circuit card." (Guthery, 43:5-11). Accordingly, as Defendants' construction provides, a terminal is the separate, external system communicating with the ICC or microcontroller.

Gemalto contends its construction is supported by the language in the specification that "terminals can be . . . any other system that communicates with an integrated circuit card or microcontroller." (Br. 30, citing 8:15-19.) But the "any other system" language actually supports Defendants' construction that the terminal is a separate system, *i.e.*, an "other system," not the same system. It in no way supports Gemalto's broader construction of "device."

[9] Neither party relies on the ordinary and customary meaning of "terminal," e.g. a train junction, the ending of a structure, etc.

Further, Gemalto's "device" could be read to encompass anything that interacts with the microcontroller, such as a mouse, a display screen, a printer, etc. That appears to be precisely Gemalto's intent, as Gemalto points to touch screens and keypads on the accused devices as being the "terminals." (Ex. J at 5 (referring to "terminal" as "an input/output device").) Here too, the named inventors expressly agreed that "a display would not be a terminal." (Guthery, 43:12-23; *see also* Wilkinson, 130:13-20.) Gemalto's construction should be rejected.

VI. *THE INTEGRATED CIRCUIT CARD* ('317 CLAIMS 58, 65, 78, 87, 88, 92)

Defendants' argument as to "the integrated circuit card" is discussed in its concurrently-filed Motion for Summary Judgment of Indefiniteness, and incorporated by reference here.

SOFTWARE TERMS

I. *HIGH LEVEL [PROGRAMMING] LANGUAGE* ('317 CLAIMS 1, 5, 31, 35, 64, 65, 84-86, 93, 94; '485 CLAIMS 7, 38, 40, 42; '727 CLAIM 3)

Defendants	*Gemalto*
programming language that must be compiled or interpreted before it can be executed by a computer	programming language that must be compiled and interpreted before it can be executed by a computer

Defendants' construction provides that a "high-level programming language" must be compiled or interpreted before being executed by a computer while Plaintiff's construction requires that it be compiled and interpreted. Gemalto previously agreed to Defendants' construction (Ex. K), but has since retracted that agreement in an obvious attempt to manufacture a validity position by seeking to limit the term to embodiments in the specification. (Br. 4-5.)

Gemalto does not dispute that Defendants' construction of "high level programming language" accords with the ordinary and customary meaning of the term. Nevertheless, Gemalto urges this Court to depart from the ordinary meaning of the term in favor of the preferred embodiment. But as Gemalto itself argues, "absent an express disclaimer by the patentee, it is

improper to limit claim terms to preferred embodiments." (Br. 29.) Gemalto does not and cannot point to any such disclaimer here, and thus its proposed construction must be rejected.

In fact, ironically, Gemalto's construction would exclude Java, the high level language that is the principal subject of the patents. Following Java's success, Sun designed computers capable of natively executing compiled Java applications, directly, without any interpretation and without the need for a virtual machine. (*See, e.g.*, Ex. L at 23("Java processors are CPUs that have been designed to execute Java bytecode instructions directly in hardware."), Ex. M ("JOP (a Java Optimized Processor) is a hardware implementation of the JVM"); *see also* Guthery, 157:8-17 (agreeing that there are no programming languages that must be both compiled and interpreted in order to be executed by a computer, including Java)). Since Java need not be interpreted before execution, Gemalto's construction would exclude Java itself, the preferred embodiment of the specification, and must therefore be rejected. *Globetrotter Software, Inc. v. Elan Computer Group, Inc.*. 362 F.3d 1367, 1381 (Fed. Cir. 2004) ("A claim interpretation that excludes a preferred embodiment from the scope of the claim "is rarely, if ever, correct." (citing *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1583 (Fed. Cir. 1996))).

II. *COMPILING / COMPILED FORM* ('317 CLAIMS 1, 31, 64-68, 73, 84-88, 91-94; '485 CLAIMS 1, 7-10, 14, 16, 21, 24, 38-43; '727 CLAIMS 1, 3-6, 10, 12, 17, 20)

Claim Term	*Defendants*	*Gemalto*
compiling	transforming a program written in a high-level programming language from source code to object code or byte code	transforming a program written in a high-level programming [language] into class files containing byte codes
compiled form	No construction necessary	class files containing byte codes

The parties agree that "compiling" a program requires "transforming" it, but dispute whether the form is limited to embodiments in the specification.

Again, Gemalto does not dispute that the ordinary and customary meaning of "compiling" accords with Defendants' construction. Indeed, Defendants' construction comes

directly from the intrinsic evidence. During prosecution, the applicants confirmed that "'compiling' is a term of art in the art of Computer Science meaning 'To transform a program written in a high-level programming language from source code to object code.'" (Ex. E at GEM3842.) This statement tracks the plain and ordinary meaning known to one of skill in the art. (*See*, Ex. N: "compiler: A program that translates all the source code of a program written in a high-level language into object code prior to execution of the program," Ex. O: "compiler: A program that reads the statements written in a human-readable programming language, such as Pascal or Modula-2, and translates the statements into a machine-readable executable program," Ex. P: "compiler: A compiler may be hardware or software. Its function is to translate the high-level computer language used by the human programmer into the machine language that is understood by the computer.")[10] The named inventors similarly agreed that compiling can create both object code and byte code. (Guthery, 104:21 to 105:9; Wilkinson, 86:24 – 87:6.)

Nevertheless, Gemalto argues "compiling" must be limited to "class file" embodiments in the specification. (Br. 5-6.) Again, however, "absent an express disclaimer by the patentee, it is improper to limit claim terms to preferred embodiments." (Br. 29.) Gemalto does not and cannot point to any such disclaimer, and thus its proposed constructions must be rejected.

In fact, the claims themselves—usually the most instructive evidence (*Phillips*, 415 F.3d at 1314)—further demonstrate that "compiled form" should not be limited to "class files" as Plaintiff suggests. For instance, some independent claims that require "compiling" explicitly require the inclusion of a "class file format," but others do not. (*Compare* Dkt. 160, Ex. A at 2 (listing claims that require a "class file format") *to id*. at 3 (listing claims that require "compiling / compiled form").) If "compiling" necessarily resulted in class files, these explicit "class file

[10] Defendants have included "or byte code," the language used by Plaintiff, as there is no objective difference between "byte code" and "object code" (see Section III, *infra*).

format" limitations would be redundant. Similarly, claims 2 and 32 of the '317 patent depend from claims that contain "compiling" and "compiled form" limitations. Yet, they further require that the high-level language to be compiled be in "a class file format." Under Gemalto's construction, these dependent claims would have the same scope as their independent claims.[11]

III. *BYTE CODES / VIRTUAL MACHINES* ('317 CLAIMS 1, 31, 58, 64, 65, 78, 84-88, 91-94; '485 CLAIMS 14, 21, 24, 38-43; '727 CLAIMS 10, 17, 20)

Claim Term	*Defendants*	*Gemalto*
byte codes / byte code format	No construction necessary	program instructions interpretable by a virtual machine
virtual machine / intermediate virtual machine	No construction necessary	programs used to interpret byte codes

Neither "byte codes" nor "virtual machine" require construction. The terms have an ordinary meaning to one of skill or art that would be understandable to a lay juror, *i.e.*, byte codes are "codes" in "byte" format for a machine or computer, while a virtual machine is a "virtual" or software implementation of a machine or computer.

Gemalto does not contend its construction of "byte codes" comports with this ordinary meaning. Instead, Gemalto concedes its construction seeks to distinguish codes executed on a virtual machine from codes executed on a real machine. (Br. 7-8.) Gemalto relies on a statement that "[w]hen a Java application is written, it is compiled into 'Class' files containing byte codes that are instructions for a hypothetical computer called a Java Virtual Machine." (1:25-28.) Even if this passage supports the notion that all Java applications are compiled into byte codes for execution on a Java Virtual Machine, which Defendants do not concede, the claims are not

[11] Gemalto also cites to statements in a pending application in the same family made *during* the present litigation that the *conversion* undertaken by a prior art reference differed from the conversion of the alleged invention. (Br. 7.) Initially, Plaintiff did not produce these materials to Defendants before its Opening Brief and they should be excluded on that basis alone. Nor should Plaintiff be permitted to manufacture "evidence" by simply reciting its opinions in a concurrent prosecution. In any event, the statements Plaintiff points to purport to distinguish prior art based on the "converting" limitation, not the "compiling" limitation. (Br. 7: "while [the Chaitlin reference] discusses 'converting' files, the conversion therein is not the same as the conversion in the present application...")

limited to Java. Rather, they claim applications written in "high level programming languages," which may or may not be executed on virtual machines. Indeed, even Gemalto concedes that byte codes are not limited to instructions for a Java Virtual Machine as disclosed in the embodiments. (Br. 8.) Moreover, named inventor Guthrey agreed that byte codes need not be interpretable by a virtual machine. (Guthery, 108:5-25.)

Furthermore, byte codes—like object codes—are simply "codes" in "byte" format that correspond to an instruction for a computer, either real or virtual. (*See, e.g.*, Dkt. 163, Ex. Q at 6:18-20.) An application is merely a sequence of these instructions, stored in a file, that is executed by a computer. Whether those instructions are executed by a real or virtual machine is not a property of the codes themselves.

For example, the same application containing Java byte codes could be executed, depending on the user's preference, on a Java Virtual Machine (virtual machine), a Java processor (real machine), or both. *See* Section I. Accordingly, Gemalto's attempt to limit the meaning of "byte codes" based on how those codes may eventually be executed is without merit. Under Gemalto's logic, the same set of Java byte code would be "byte code" when executed by a Java Virtual Machine and not "byte code" when executed by a Java processor.

IV. *CLASS FILE FORMAT* ('317 CLAIMS 58, 63, 67, 87; '485 CLAIMS 1, 6, 9; '727 CLAIMS 1, 2, 5)

Claim Term	*Defendants*	*Gemalto*
class file format	a file format containing byte codes that are instructions for a hypothetical computer or virtual machine	a file format containing byte codes that are instructions for a virtual machine
application having a class file format	No construction necessary	a program whose compiled form is a class file format

The parties agree that a "class file format" contains byte codes that correspond to instructions. The parties dispute whether those byte codes must be executed on a "virtual machine" (Plaintiff) or whether they can be executed on real computers as well (Defendants).

As discussed above, byte codes can be executed on both virtual machines (*e.g.*, Java Virtual Machines) and real computers (*e.g.*, Java processors). Defendants' use of "hypothetical computer" is direct from the specification, as even Gemalto concedes. (Br. 8; *see also* 1:24-27: "When a Java application is written, it is compiled into "Class" files containing byte codes that are instructions for a hypothetical computer called a Java Virtual Machine.") Defendants' construction allows for the possibility that the "hypothetical computer" be implemented as a real computer. Gemalto's use of "virtual machine" alone appears to foreclose that possibility. Defendants' construction is therefore more accurate and should be adopted.

The term "application having a class file format" has no special meaning, given a construction for "class file format." A lay jury would understand what an "application" is. Thus no construction is necessary. Nonetheless, Gemalto insists on asserting a construction for this term, claiming it to be "consistent with the [] claims of the Gemalto Patents." (Br. 9.) Tellingly, Gemalto fails to even recite the claims, since that language is fatal to its construction. For example, claim 58 of the '317 patent recites "a derivative application derived from an application having a class file format wherein the application is derived from an application having a class file format by first compiling the application having a class file format into a compiled form." And claim 2 of the '317 patent states that "the high level programming language format comprises a class file format." Thus as recited in the claims, the "class file format" is the application's form before compiling— when it is written in a high level programming language format—not its compiled form. Gemalto's attempt to rewrite claim language should be rejected.

V. *CONVERTING / CONVERTED FORM* ('317 CLAIMS 1, 31, 58, 64, 78, 84-88, 91-94; '485 CLAIMS 1, 21, 38, 42; '727 CLAIMS 1, 10, 17, 20)

Claim Term	*Defendants*	*Gemalto*
converting	No construction necessary	postprocessing the byte codes of the compiled form into a converted form
converted form	No construction	byte codes interpretable by a different virtual

	necessary	machine than the compiled form

Neither "converting" nor "converted form" have any special meaning in the art or in the patents. Each of the claims states what functionality—if any—is required by the converter in that claim. A lay jury would understand that "converting" requires the transforming from one form to another, and that a "converted form" is merely the result of "converting."

Gemalto does not contend that these terms have a different meaning to those of skill in the art. Nevertheless, Gemalto appends various unnecessary or extraneous limitations to the ordinary meaning of "converting," such as "postprocessing," "compiled form," and "converted form." The resulting redundancy becomes more apparent when Gemalto's construction is substituted into the claim language, such as claim 1 of the '317 patent:

> an application derived from a program written in a high level programming language format wherein the application is derived from a program written in a high level programming language format by first compiling the program into a compiled form and then ~~converting~~ *postprocessing the byte codes of the compiled form into a converted form* the compiled form into a converted form

Further, certain claims already require that "converting" occur after "compiling" and that it change a "compiled form" into a "converted form." (*See, e.g.*, '727 Patent claims 1, 3.) Gemalto's proposed construction improperly reads limitations from the specification and already in some claims into <u>all</u> the claims, and should thus be rejected.

Gemalto attempts a similar tactic with "converted form." Here, it adds limitations that require "byte codes," "virtual machines," and that the compiled form be interpretable by a <u>different</u> virtual machine. Gemalto fails to point to any disclaimer in the specification that would limit the plain and ordinary meaning of "converted form." (Br. 29-30.) Further, while some claims require that the converted form be in a byte code format—*e.g.* claim 14 of the '485 patent—other claims do not, *e.g.*, claims 7 and 10 of the '485 patent, from which claim 14 depends. Similarly, some claims require that the converted form be interpreted by a "virtual

machine," whereas others require that the converted form be interpreted by an "interpreter." (*See* claims 17 and 3 of the '727 patent, respectively.) And only six[12] of the twenty-four asserted independent claims suggest that the compiled form need be "interpretable" at all, let alone that the compiled form be interpreted by a different virtual machine than the converted form. Gemalto's attempt to import limitations from the specification that appear in only some of the claims into all of the claims should be rejected. *Sta-Rite Industries, LLC v. ITT Corp.*, 682 F.Supp.2d 738, 743 (E.D.Tex. 2010) ("Courts presume a difference in meaning and scope when a patentee uses different phrases in separate claims.")

VI. *SPECIFIC BYTE CODES / GENERAL BYTE CODES* ('317 CLAIMS 1, 31, 58, 64, 65, 78, 84-87, 91, 94; '485 CLAIMS 14, 24, 39, 41, 43; '727 CLAIMS 10, 20)

Claim Term	*Defendants*	*Gemalto*
specific byte codes / generic byte codes	smaller byte codes / larger byte codes for the same operation	*specific byte code*: a byte code with a built-in argument or operand *generic byte code*: a byte code with a separate, accompanying argument or operand

Defendants' construction captures the essence of this element: a smaller number of byte codes is replaced with a functionally-equivalent larger number of byte codes. (Br. 12, explaining that the "ILOAD_0" byte code is replaced with the "ILOAD" and "0" byte codes.) Gemalto's use of "argument or operand" is likely to confuse a lay jury, thereby defeating the purpose of claim construction. As Gemalto does not contend that Defendants' construction is incorrect, and as that construction will not confuse a lay jury, Defendants' construction should be adopted.

MEANS PLUS FUNCTION CLAIMS

Defendants' arguments as to the means plus function limitations are discussed in its concurrently-filed Motion for Summary Judgment of Indefiniteness, and incorporated by reference here. One additional term, "attributes," also appears in a means-plus-function claim.

[12] Specifically, claims 78, 88, 91, and 92 of the '317 patent, claim 21 of the '485 patent, and claim 17 of the '727 patent

VII. *ATTRIBUTES* ('317 CLAIM 66; '485 CLAIM 8; '727 CLAIM 4)

Defendants	*Gemalto*
Data structures having the following general format: attibute_info { u2 attribute_name_index; u4 attribute _length; u1 info[attribute_length]; }	No construction necessary. Alternate: information in a class file

The term "attributes" appears in one asserted means-plus-function claim of each patent. (Dkt. 169, Ex. A at 1.) The parties agree that the corresponding structure is a "computer programmed to perform the algorithm of element 51c of Figure 5 and 9:62 to 10:5 of the specification." (Dkt 169 at 2.) This portion of the specification states that "A Java class file 24a stores information pertaining to the byte codes in the class file in the Attributes section 44 of the Java class file." (9:64-66.) Since the term "attributes" appears in a means-plus-function element, it is bound by the corresponding structure described in the specification: "the Attributes section [] of the Java class file." Gemalto agrees that Defendants' construction describes that structure. (Br. 11). Thus, Defendants' construction governs.

Instead, Gemalto apparently intends to broaden this term to encompass any data structure in a class file. But because it appears solely in the context of means-plus-function limitations, such an approach is impermissible as a matter of law. *WMS Gaming, Inc. v. International Game Technology*, 184 F.3d 1339, 1348-49 (Fed. Cir. 1999). Further, named inventor Guthery agreed that an "attribute" is an agreed-upon structure, and that "there is data in a class file that is not characterized as being in an attribute structure." (Guthery, 109:2 – 110:7.) Gemalto's construction would thus include all "information in a class file," including data which is not an "attribute." Gemalto's overbroad construction must be rejected.

CONCLUSION

For the foregoing reasons, Defendants respectfully ask the Court to adopt their proposed constructions and to reject the constructions proposed by Gemalto.

Dated this 2nd day of April, 2012.

By /s/ David Perlson

QUINN EMANUEL URQUHART & SULLIVAN, LLP
Charles K. Verhoeven
charlesverhoeven@quinnemanuel.com
David A. Perlson
davidperlson@quinnemanuel.com
Sean Pak
seanpak@quinnemanuel.com
50 California Street, 22nd Floor, San Francisco, California 94111-4788
Telephone: (415) 875-6600
Facsimile: (415) 875-6700

Michael E. Jones
State Bar No. 10929400
mikejones@potterminton.com
Allen F. Gardner
State Bar No. 24043679
allengardner@potterminton.com
POTTER MINTON, P.C.
A Professional Corporation
110 N. College Ave., Suite 500
Tyler, Texas 75702
Tel: (903) 597-8311
Fax: (903) 593-0846

Attorneys for Defendants HTC Corporation, HTC America, Inc., Exedea, Inc., Samsung Electronics Co., Ltd., Samsung Telecommunications America LLC, Motorola Mobility, Inc., and Google Inc.

CERTIFICATE OF SERVICE

I hereby certify that counsel of record who are deemed to have consented to electronic service are being served this 2nd day of April, 2012, with a copy of this document via the Court's CM/ECF system per Local Rule CV-5(a)(3).

By /s/ Antonio Sistos